KU LEUVEN

STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics

Citation/Reference	Authors: Paul Didier, Toon van Waterschoot, Simon Doclo, Marc Moonen Title: Sampling Rate Offset Estimation and Compensation for Distributed Adaptive Node-Specific Signal Estimation in Wireless Acoustic Sensor Networks Journal/Conference/Book: IEEE Open Journal of Signal Processing (via IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2023 track) Vol/Book Chapter: N/A Pages: N/A ISBN: N/A
Dates **Archived version**	**Conference paper/Abstract** • Date of conference: June 4-9, 2023 ■ Submitted version ☐ Accepted (Author manuscript: the content is identical to the content of the published paper, but without the final typesetting by the publisher) ☐ Published ☐ Published (Open Access)
Acknowledgements	In this manuscript, the following funding is acknowledged: ■ EU H2020 ☐ EU Horizon Europe ☐ FWO ☐ KULeuven Internal Funds ☐ VLAIO ☐ AI ☐ Other
Published version	N/A
Copyright statement	This research work was carried out in the frame of the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 956369: "Service-Oriented Ubiquitous Network-Driven Sound — SOUNDS". The research leading to these results has received funding from the European Research Council under the European Union's Horizon 2020 research and innovation program / ERC Consolidator Grant: SONORA (no. 773268). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information. The scientific responsibility is assumed by the authors.
Author contact	paul.didier@esat.kuleuven.be

IEEE Open Journal of Signal Processing

Sampling Rate Offset Estimation and Compensation for Distributed Adaptive Node-Specific Signal Estimation in Wireless Acoustic Sensor Networks

PAUL DIDIER[1], TOON VAN WATERSCHOOT[1], SIMON DOCLO[2], AND MARC MOONEN[1]

[1]STADIUS Center for Dynamical Systems, Department of Electrical Engineering (ESAT), KU Leuven, 3001 Leuven, Belgium
[2]Signal Processing Group, Department of Medical Physics and Acoustics and Cluster of Excellence Hearing4all, University of Oldenburg, Oldenburg, Germany

CORRESPONDING AUTHOR: Paul Didier (e-mail: paul.didier@esat.kuleuven.be).

This research work was carried out in the frame of the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 956369: "Service-Oriented Ubiquitous Network-Driven Sound — SOUNDS". The research leading to these results has received funding from the European Research Council under the European Union's Horizon 2020 research and innovation program / ERC Consolidator Grant: SONORA (no. 773268). This paper reflects only the authors' views and the Union is not liable for any use that may be made of the contained information. The scientific responsibility is assumed by the authors.

ABSTRACT Sampling rate offsets (SROs) between devices in a heterogeneous wireless acoustic sensor network (WASN) can hinder the ability of distributed adaptive algorithms to perform as intended when they rely on coherent signal processing. In this paper, we present an SRO estimation and compensation method to allow the deployment of the distributed adaptive node-specific signal estimation (DANSE) algorithm in WASNs composed of asynchronous devices. The signals available at each node are first utilised in a coherence-drift-based method to blindly estimate SROs which are then compensated for via phase shifts in the frequency domain. A modification of the weighted overlap-add (WOLA) implementation of DANSE is introduced to account for SRO-induced full-sample drifts, permitting per-sample signal transmission via an approximation of the WOLA process as a time-domain convolution. The performance of the proposed algorithm is evaluated in the context of distributed noise reduction for the estimation of a target speech signal in an asynchronous WASN.

INDEX TERMS Sampling rate offsets, coherence drift, signal enhancement, weighted overlap-add, wireless acoustic sensor networks

I. Introduction

Wireless acoustic sensor networks (WASNs) have been a subject of great interest in recent years as they provide a number of advantages over centralised systems performing audio signal processing tasks [1]. Novel algorithmic solutions aim to utilise the increased flexibility and scalability of WASNs in order to tackle various audio signal processing challenges in a distributed fashion, bypassing the need for a data fusion centre with which all nodes communicate.

This paper focuses on distributed signal estimation, where each node in the WASN aims to estimate a node-specific desired signal. The distributed adaptive node-specific signal estimation (DANSE) algorithm was originally formulated in [2], [3] to tackle this problem. This algorithm is designed to allow each node of a fully connected WASN to achieve centralised performance upon convergence by iteratively computing its own multichannel Wiener filter (MWF) while only exchanging single-channel signals with other nodes. Although the DANSE algorithm has been tested under various conditions and for different tasks [4], a key aspect allowing its robust deployment in real-world scenarios has yet to be addressed, namely, signals asynchronicity.

In many practical applications such as teleconferencing systems or smart domotics, the WASN is heterogeneous, i.e., composed of various interconnected devices such as laptops, tablets, or hearing aids. Each device samples the incoming acoustic information at a specific rate via its own analog-to-digital converter based on an internal clock, the sampling rate of which may differ from the nominal value provided by the manufacturer [5]. The sampling rate mismatch between two devices can be quantified as the sampling rate offset (SRO), generally expressed in parts-per-million (PPM). SROs in the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

range of ± 500 PPM have been measured between commonly used devices and reported in [5]. The same study showed that SROs can slowly vary through time, e.g., when the devices undergo significant temperature changes.

SROs lead to an increasing time-drift between signals sampled by different clocks, which inhibits their use in algorithms that rely on coherent signal processing [6]. Notably, the performance of signal enhancement algorithms based on the MWF such as the DANSE algorithm depends on the computation of accurate spatial covariance matrices. DANSE can thus be expected to be sensitive to a lack of synchronicity between locally recorded microphone signals and signals received from other nodes. In fact, literature around DANSE has so far assumed that all nodes involved in the algorithm have exactly the same sampling rate [2], [3], [7]–[9].

In this paper, we propose a methodology to relax the synchronicity assumption, bringing DANSE closer to robust deployment in real-life scenarios. The presence of SROs is addressed in a fully connected WASN where node and source positions are fixed. Time-invariant SROs are considered, assuming that no temperature changes occur during the convergence phase of the algorithm. Per-node-pair SRO estimation is performed blindly based on a coherence-drift method [10]–[12]. The weighted overlap-add (WOLA) implementation of the generalised eigenvalue decomposition (GEVD) DANSE algorithm [13] is modified to permit detection of full-sample drifts (FSDs) via per-sample signal broadcasting. This is achieved by approximating the WOLA process used for local signal fusion (analysis, filtering in the short-time Fourier transform (STFT) domain, and synthesis) as a single time-domain convolution operation. This method allows to retain the low complexity of WOLA processing for the more costly steps of GEVD-based filter update and desired signal estimation. The estimated SROs and the detected FSDs are then compensated for via phase shifts in the STFT-domain. The performance of the proposed algorithm is evaluated in the context of distributed noise reduction for the estimation of a target speech signal.

The paper is organised as follows. In Section II, the centralised GEVD-MWF is reviewed. The key aspects of the theory and implementation of the DANSE algorithm are summarised in Section III. The proposed method for SRO estimation and compensation within the DANSE framework is presented in detail in Section IV. The performance of the proposed method is then analysed by means of simulations in asynchronous WASNs in Section V. Finally, conclusive remarks are formulated in Section VI.

II. GEVD-MWF-based signal estimation

A WASN composed of K nodes is considered, where each node $k \in \mathcal{K} = \{1, \dots, K\}$ has $M_k \geq 1$ microphones. The total number of microphones in the network is denoted by $M = \sum_{k \in \mathcal{K}} M_k$. In the acoustic scene, one localised static desired signal source (e.g., a talker) and $J \geq 1$ localised static noise sources are present. The signals recorded by node k can be represented in the STFT domain at frame i and frequency bin κ via an additive-noise signal model:

$$\mathbf{y}_k[\kappa, i] = \mathbf{s}_k[\kappa, i] + \mathbf{n}_k[\kappa, i], \tag{1}$$

where $\mathbf{y}_k[\kappa, i]$, $\mathbf{s}_k[\kappa, i]$, and $\mathbf{n}_k[\kappa, i]$ are M_k-dimensional vectors corresponding to the microphone signals, the desired signal components of these signals, and the noise components, respectively. The additive noise is assumed to be uncorrelated with the desired signal.

In centralised processing, the signal vector available at the fusion centre is defined as an M-dimensional stacked version $\mathbf{y}[\kappa, i] = \left[\mathbf{y}_1^T[\kappa, i], \dots, \mathbf{y}_K^T[\kappa, i]\right]^T$ of the node-specific microphone signals where $(\cdot)^T$ denotes the transpose operation. Similarly to (1), this vector can be expressed as $\mathbf{y}[\kappa, i] = \mathbf{s}[\kappa, i] + \mathbf{n}[\kappa, i]$ with $\mathbf{n}[\kappa, i] = \left[\mathbf{n}_1^T[\kappa, i], \dots, \mathbf{n}_K^T[\kappa, i]\right]^T$ and $\mathbf{s} = \left[\mathbf{s}_1^T[\kappa, i], \dots, \mathbf{s}_K^T[\kappa, i]\right]^T$.

The objective of node k is then to estimate a local desired signal $d_k[\kappa, i]$ based on $\mathbf{y}[\kappa, i]$. Define without loss of generality (w.l.o.g.) the desired signal $d_k[\kappa, i]$ at node k to be the desired signal component of the first local microphone signal, i.e., $d_k[\kappa, i] = \mathbf{e}_{d_k}^T \mathbf{s}[\kappa, i]$, where $\mathbf{e}_{d_k}$ selects the appropriate channel of $\mathbf{s}[\kappa, i]$. An optimal filter $\bar{\mathbf{w}}_k[\kappa, i]$ can be obtained by minimising the mean squared error (MSE) between the desired signal and the filtered microphone signals:

$$\bar{\mathbf{w}}_k[\kappa, i+1] = \underset{\mathbf{w}_k[\kappa]}{\arg\min} E\left\{ \left| d_k[\kappa, i] - \mathbf{w}_k^H[\kappa] \mathbf{y}[\kappa, i] \right|^2 \right\}, \tag{2}$$

where $(\cdot)^H$ denotes complex conjugation and $E\{\cdot\}$ the expected value operation. The closed-form solution of (2) is the MWF:

$$\bar{\mathbf{w}}_k[\kappa, i+1] = \left(\bar{\mathbf{R}}_{\mathbf{yy}}[\kappa, i]\right)^{-1} \bar{\mathbf{R}}_{\mathbf{ss}}[\kappa, i] \mathbf{e}_{d_k}, \tag{3}$$

where $\bar{\mathbf{R}}_{\mathbf{yy}}[\kappa, i] = E\{\mathbf{y}[\kappa, i]\mathbf{y}^H[\kappa, i]\}$ is the network-wide microphone signal covariance matrix and $\bar{\mathbf{R}}_{\mathbf{ss}}[\kappa, i] = \bar{\mathbf{R}}_{\mathbf{yy}}[\kappa, i] - \bar{\mathbf{R}}_{\mathbf{nn}}[\kappa, i]$, where $\bar{\mathbf{R}}_{\mathbf{nn}}[\kappa, i] = E\{\mathbf{n}[\kappa, i]\mathbf{n}^H[\kappa, i]\}$ is the network-wide noise-only covariance matrix. Assuming short-term stationarity of the signals, the covariance matrices can be estimated by averaging over observations of $\mathbf{y}[\kappa, i]\mathbf{y}^H[\kappa, i]$. In a speech enhancement scenario with stationary noise, the on-off behaviour of the desired signal can be exploited via a voice activity detector (VAD) [14], [15] to isolate noise-only observations of $\mathbf{y}[\kappa, i]$. The estimation of $\bar{\mathbf{R}}_{\mathbf{yy}}[\kappa, i]$ and $\bar{\mathbf{R}}_{\mathbf{nn}}[\kappa, i]$ can then be performed via exponential averaging:

$$\begin{aligned} &\text{VAD = 1 :} \\ &\quad \hat{\mathbf{R}}_{\mathbf{yy}}[\kappa, i] = \beta \hat{\mathbf{R}}_{\mathbf{yy}}[\kappa, i-1] + (1-\beta)\mathbf{y}[\kappa, i]\mathbf{y}^H[\kappa, i], \\ &\text{VAD = 0 :} \\ &\quad \hat{\mathbf{R}}_{\mathbf{nn}}[\kappa, i] = \beta \hat{\mathbf{R}}_{\mathbf{nn}}[\kappa, i-1] + (1-\beta)\mathbf{y}[\kappa, i]\mathbf{y}^H[\kappa, i], \end{aligned} \tag{4}$$

where the real-valued number β acts as a forgetting factor, $0 \ll \beta \leq 1$, typically chosen close to 1 to preserve spatial coherence between microphone signals [1].

In the presence of a single desired signal source, the signal model implies that $\hat{\mathbf{R}}_{\mathbf{ss}}[\kappa,i]$ should be a rank-1 matrix [16]. However, the estimated $\hat{\mathbf{R}}_{\mathbf{ss}}[\kappa,i]=\hat{\mathbf{R}}_{\mathbf{yy}}[\kappa,i]-\hat{\mathbf{R}}_{\mathbf{nn}}[\kappa,i]$ generally has a rank larger than 1. A GEVD-based approach was proposed in [16] to obtain a rank-1 approximation of $\hat{\mathbf{R}}_{\mathbf{ss}}[\kappa,i]$. The GEVD of the matrix pencil $\{\hat{\mathbf{R}}_{\mathbf{yy}}[\kappa,i],\hat{\mathbf{R}}_{\mathbf{nn}}[\kappa,i]\}$ yields:

$$\begin{aligned}\hat{\mathbf{R}}_{\mathbf{yy}}[\kappa,i]&=\hat{\mathbf{Q}}[\kappa,i]\hat{\mathbf{\Sigma}}[\kappa,i]\hat{\mathbf{Q}}^{H}[\kappa,i],\\ \hat{\mathbf{R}}_{\mathbf{nn}}[\kappa,i]&=\hat{\mathbf{Q}}[\kappa,i]\hat{\mathbf{Q}}^{H}[\kappa,i],\end{aligned}\tag{5}$$

with $\hat{\mathbf{Q}}[\kappa,i]$ an $M\times M$ matrix of which the columns are the generalised eigenvectors (GEVCs) and $\hat{\mathbf{\Sigma}}[\kappa,i]$ is a diagonal matrix of which the diagonal elements are the corresponding generalised eigenvalues (GEVLs). The GEVLs in $\hat{\mathbf{\Sigma}}[\kappa,i]$ are assumed to be ordered by decreasing magnitude. A rank-1 approximation of $\hat{\mathbf{R}}_{\mathbf{ss}}[\kappa,i]$ can then be obtained by using (5) and nullifying the $M-1$ smallest GEVLs. Substituting into (3) then leads to the GEVD-MWF:

$$\hat{\mathbf{w}}_k[\kappa,i+1]=\hat{\mathbf{Q}}^{-H}[\kappa,i]\mathbf{\Lambda}[\kappa,i]\hat{\mathbf{Q}}^{H}[\kappa,i]\mathbf{e}_{d_k},\tag{6}$$

with $\mathbf{\Lambda}[\kappa,i]=\mathrm{diag}\{1-1/\hat{\sigma}_1[\kappa,i],0,\ldots,0\}$, where $\mathrm{diag}\{\cdot\}$ transforms a vector into a diagonal matrix and $\hat{\sigma}_1[\kappa,i]$ is the largest GEVL. Finally, the desired signal at frequency bin κ and frame i is estimated as $\hat{d}_k[\kappa,i]=\hat{\mathbf{w}}_k^H[\kappa,i+1]\mathbf{y}[\kappa,i]$.

III. The DANSE algorithm

The DANSE algorithm [2] provides a distributed implementation of the MWF described in Section II as an adaptive algorithm where nodes iteratively updates their local filter estimates. Different node-updating schemes exist: (i) sequential updating [2], (ii) simultaneous updating [3], and (iii) asynchronous updating [3]. Strategies (i) and (ii) rely on a network-wide update protocol that coordinates the updates, unlike strategy (iii) [3]. Since the presence of unknown SROs between nodes challenges the deployment of a coordination protocol, asynchronous updating is assumed in the following. To avoid limit cycles due to asynchronous updating [3], relaxed filter updates can be performed [17]. Computational delays due to data transmission, reception, and processing are assumed to be negligible in this paper.

As described in [13], the DANSE algorithm can be implemented using weighted overlap-add (WOLA) processing to efficiently perform short-time Fourier analysis and synthesis [18]. Using WOLA, time-domain microphone signals are processed on a frame-by-frame basis. WOLA analysis consists in applying an N-point DFT to a windowed frame of a time-domain signal, with the frame size equal to the DFT size, effectively transforming the time-domain signal frame into an STFT-domain signal frame. In DANSE, all filtering can be conducted in the STFT domain, resulting in a lower computational complexity as compared to a time-domain implementation [13]. Each new WOLA frame then corresponds to a new DANSE iteration where the nodes update their filter estimate. As a WOLA implementation of DANSE is assumed in the following, the variable i simultaneously denotes the STFT frame index as well as the DANSE iteration index, i.e., nodes update their filter estimates at each new i.

Although a variety of network topologies can exist, a fully connected WASN is assumed in this paper. All the nodes that can communicate with node k are grouped in the set $\mathcal{K}_k=\mathcal{K}\backslash\{k\}$. The DANSE algorithm in a fully connected WASN operates in two main stages: signals fusion and broadcasting on the one hand, and filters updates on the other hand [2]. At each frame i, each node k fuses its M_k local microphone signals into a single-channel signal $z_k[\kappa,i]\ \forall\kappa\in\{1\ldots N\}$ using a fusion vector $\mathbf{p}_k[\kappa,i]$ before broadcasting it to the other nodes, which reduces the per-node communication cost by a factor M_k (i.e., a factor M/K over the entire network) compared to the centralised MWF of Section II. An appropriate definition of $\mathbf{p}_k[\kappa,i]$ guarantees convergence of the DANSE algorithm to the centralised MWF solution [2]. In the WOLA implementation, a time-domain fused signal denoted by $\dot{z}_k[n]$ is obtained via WOLA synthesis (inverse DFT followed by windowing) and overlap-add of the fused signal frames $z_k[\kappa,i]$, where n denotes the sample index. The time-domain signal $\dot{z}_k[n]$ is then broadcast to other nodes, as summarised in Algorithm 1.

The STFT-domain signals available at node k at iteration i are grouped into the vector:

$$\tilde{\mathbf{y}}_k[\kappa,i]=\left[\mathbf{y}_k^T[\kappa,i]\mid\mathbf{z}_{-k}^T[\kappa,i]\right]^T\tag{7}$$

where $\mathbf{y}_k[\kappa,i]$ contains the local microphone signals and $\mathbf{z}_{-k}[\kappa,i]$ is a stacked version of all the microphone signals received from other nodes. As in the centralised case (cfr. (1)), $\tilde{\mathbf{y}}_k[\kappa,i]$ can be written as a sum of a desired signal component $\tilde{\mathbf{s}}_k[\kappa,i]$ and a noise component $\tilde{\mathbf{n}}_k[\kappa,i]$. Node k aims to compute the i-th STFT frame of its desired signal estimate $\hat{d}_k[\kappa,i]$ via multichannel filtering of $\tilde{\mathbf{y}}_k[\kappa,i]$. The filter at node k is denoted by $\tilde{\mathbf{w}}_k[\kappa,i]=[\mathbf{w}_{kk}^T[\kappa,i]\mid\mathbf{g}_{k-k}^T[\kappa,i]]^T$, where $\mathbf{w}_{kk}[\kappa,i]$ is applied to the local microphone signals $\mathbf{y}_k[\kappa,i]$ and $\mathbf{g}_{k-k}[\kappa,i]$ is applied to the fused microphone signals $\mathbf{z}_{-k}[\kappa,i]$. The filter at node k at iteration $i+1$ is obtained by minimising the MSE between the desired signal and its estimate:

$$\tilde{\mathbf{w}}_k[\kappa,i+1]=\underset{\mathbf{w}_k[\kappa]}{\arg\min}\ E\left\{\left|d_k[\kappa,i]-\mathbf{w}_k^H[\kappa]\tilde{\mathbf{y}}_k[\kappa,i]\right|^2\right\},\tag{8}$$

and the i-th STFT frame of desired signal estimate is then $\hat{d}_k[\kappa,i]=\tilde{\mathbf{w}}_k^H[\kappa,i+1]\tilde{\mathbf{y}}_k[\kappa,i]$. Equation (8) has the same structure as (2), be it with a different definition of the filter $\mathbf{w}_k[\kappa]$ and input vector $\tilde{\mathbf{y}}_k[\kappa,i]$, hence its solution again corresponds to an MWF. With the covariance matrices $\tilde{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}[\kappa,i]$ and $\tilde{\mathbf{R}}_{\mathbf{n}_k\mathbf{n}_k}[\kappa,i]$ defined and estimated per node instead of centrally as in (4), a GEVD is applied to the matrix pencil $\{\tilde{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}[\kappa,i],\tilde{\mathbf{R}}_{\mathbf{n}_k\mathbf{n}_k}[\kappa,i]\}$ and, similarly to (6), the filter is computed as:

$$\tilde{\mathbf{w}}_k[\kappa, i+1] = \tilde{\mathbf{Q}}_k^{-H}[\kappa, i]\tilde{\mathbf{\Lambda}}_k[\kappa, i]\tilde{\mathbf{Q}}_k^{H}[\kappa, i]\mathbf{e}_{d_k}, \quad (9)$$

where, at frequency κ and iteration i, $\tilde{\mathbf{Q}}_k[\kappa, i]$ is an $(M_k + K - 1) \times (M_k + K - 1)$ matrix of which the columns are the GEVCs, $\tilde{\mathbf{\Lambda}}_k[\kappa, i] = \text{diag}\{1 - 1/\tilde{\sigma}_{k1}[\kappa, i], 0, \ldots, 0\}$ and $\tilde{\sigma}_{k1}[\kappa, i]$ is the largest GEVL. Finally, to ensure convergence of $\mathbf{w}_{kk}[\kappa, i]$ towards the corresponding elements of the centralised MWF, the fusion vector at iteration i is defined as $\mathbf{p}_k[\kappa, i] = \mathbf{w}_{kk}[\kappa, i]$, such that:

$$z_k[\kappa, i] = \mathbf{w}_{kk}^H[\kappa, i]\mathbf{y}_k[\kappa, i]. \quad (10)$$

The WOLA implementation of the DANSE algorithm is summarised in Algorithm 1, where the M_k local time-domain microphone signals at node k are denoted by $\dot{\mathbf{y}}_k[n]$. The time-domain signal obtained after WOLA synthesis and overlap-add of consecutive frames $\hat{d}_k[\kappa, i]$ is denoted by $\dot{\hat{d}}_k[n]$. The time-domain fused signals are grouped in the vector $\dot{\mathbf{z}}_{-k}[n] = \left[\dot{z}_1[n] \ \ldots \ \dot{z}_{k-1}[n] \ \dot{z}_{k+1}[n] \ \ldots \ \dot{z}_K[n]\right]^T$. The WOLA window shift, corresponding to the number of new samples recorded between two consecutive DANSE iterations, is denoted by N_s.

Algorithm 1 WOLA-based DANSE in a synchronised, fully connected WASN (50% window overlap).

1: Initialise $\tilde{\mathbf{w}}_k[\kappa, 0] \ \forall \ k \in \mathcal{K}$;
2: Each node $k \in \mathcal{K}$ performs the following, starting simultaneously;
3: **for** $i = 1, 2, 3, \ldots$ **do**
4: Record N_s new samples of $\dot{\mathbf{y}}_k[n]$ since $i - 1$;
5: WOLA analysis on N most recent $\dot{\mathbf{y}}_k[n]$ samples to obtain $\mathbf{y}_k[\kappa, i]$;
6: Perform signal fusion via (10) to obtain $z_k[\kappa, i]$;
7: WOLA synthesis on $z_k[\kappa, i]$ and overlap-add with previous frame to obtain N_s new $\dot{z}_k[n]$ samples;
8: Transmit N_s most recent $\dot{z}_k[n]$ samples to $\mathcal{K}_k$;
9: Build $\dot{\mathbf{z}}_{-k}[n]$ from samples received from $\mathcal{K}_k$;
10: WOLA analysis on N most recent $\dot{\mathbf{z}}_{-k}[n]$ samples to obtain $\mathbf{z}_{-k}[\kappa, i]$;
11: Build $\tilde{\mathbf{y}}_k[\kappa, i] = [\mathbf{y}_k^T[\kappa, i] \mid \mathbf{z}_{-k}^T[\kappa, i]]^T$;
12: Compute $\tilde{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}[\kappa, i]$ and $\tilde{\mathbf{R}}_{\mathbf{n}_k\mathbf{n}_k}[\kappa, i]$ via (4);
13: Compute the filter estimates $\tilde{\mathbf{w}}_k[\kappa, i+1]$ via (9);
14: Compute new frame $\hat{d}_k[\kappa, i]$;
15: WOLA synthesis on $\hat{d}_k[\kappa, i]$ and overlap-add with the previous frame to build $\dot{\hat{d}}_k[n]$.
16: **end for**

In the presence of SROs, the time misalignments between the local microphone signals and the fused microphone signals from other nodes lead to incorrect covariance matrix updates which, in turn, inhibit the computation of useful filter estimates via (9). In the following section, we propose a method for SRO estimation and compensation applicable to the WOLA implementation of the DANSE algorithm.

IV. SRO estimation and compensation

The SRO between node k and q is denoted by ε_{kq} such that $f_{\text{s},q} = f_{\text{s},k}(1+\varepsilon_{kq})$, where $f_{\text{s},k}$ and $f_{\text{s},q}$ are the sampling rate of node k and q, respectively. In the following, it is assumed that the SROs are time-invariant and that signals recorded by the same node are synchronised. Although a fully connected WASN is assumed, this SRO estimation and compensation method can be also adopted in other network topologies.

A. Coherence-drift-based SRO estimation

In order to allow any node $k \in \mathcal{K}$ to blindly estimate the SROs $\{\varepsilon_{kq}\}_{q\in\mathcal{K}_k}$ using the signals it can access in the DANSE algorithm, we use a coherence-drift method based on principles introduced in [10] and [12]. At frame i and at node k, considering one other node $q \in \mathcal{K}_k$, the available STFT-domain signals are (i) the local microphone signals $\mathbf{y}_k[\kappa, i]$ and (ii) the received fused signal $z_q[\kappa, i]$. The first local microphone signal $y_{k,1}[\kappa, i]$ is used w.l.o.g. in the following.

The sampling rate mismatch can simply be approximated in the STFT-domain via the linear phase drift (LPD) model [19], [20] at any frequency bin $\kappa \in \{1, \ldots, N\}$, i.e.:

$$z_q[\kappa, i] \approx \check{z}_q[\kappa, i] \cdot \exp\left(\text{j}\frac{2\pi}{N}\kappa\varepsilon_{kq}[i]N_\text{c}[i]\right), \quad (11)$$

where $N_\text{c}[i]$ is the central sample index of frame i and $\check{z}_q[\kappa, i]$ is the $\varepsilon_{kq}[i]$-compensated version of $z_q[\kappa, i]$ (synchronised with $y_{k,1}[\kappa, i]$). The product $\tau_{kq}[i] = \varepsilon_{kq}[i]N_\text{c}[i]$ is the average accumulated time-drift between $z_q[\kappa, i]$ and $y_{k,1}[\kappa, i]$. The LPD model relies on the assumption that the SRO-induced time drift is constant within one frame, implying that the model best approximates the effect of SROs for small $\varepsilon_{kq}[i]$.

Based on (11), $\varepsilon_{kq}[i]$ can be estimated by node k as follows. First, we define the instantaneous estimate of the cross-power spectral density (PSD) $\Psi_{kq}[\kappa, i]$ as $\Psi_{kq}[\kappa, i] = y_{k,1}[\kappa, i] \cdot z_q^*[\kappa, i]$, where $\cdot^*$ denotes complex conjugation. Similarly, the instantaneous auto-PSD estimates are defined as $\Psi_{kk}[\kappa, i] = |y_{k,1}[\kappa, i]|^2$ and $\tilde{\Psi}_{qq}[\kappa, i] = |z_q[\kappa, i]|^2$. An instantaneous estimate of the coherence between $y_{k1}[\kappa, i]$ and $z_q[\kappa, i]$ can then be obtained as:

$$\Gamma_{kq}[\kappa, i] = \frac{\Psi_{kq}[\kappa, i]}{\sqrt{\Psi_{kk}[\kappa, i] \cdot \tilde{\Psi}_{qq}[\kappa, i]}}. \quad (12)$$

The SRO can now be estimated by defining the product $P_{\Gamma,kq}[\kappa, i]$ between the instantaneous coherence estimate at frame i and at frame $i - l_\text{d}$ as:

$$P_{\Gamma,kq}[\kappa, i] = \Gamma_{kq}[\kappa, i] \cdot \Gamma_{kq}^*[\kappa, i - l_\text{d}]. \quad (13)$$

Based on the LPD model and assuming static sources, it can be shown that an SRO estimate $\hat{\varepsilon}_{kq}[i]$ proportional to the phase of $P_{\Gamma,kq}[\kappa, i]$ [10], [12] is obtained as:

$$\angle\{P_{\Gamma,kq}[\kappa, i]\} = \frac{2\pi}{N}\kappa l_\text{d} N_\text{s}\hat{\varepsilon}_{kq}[i], \quad (14)$$

where $\angle\{\cdot\}$ denotes the phase. Increasing the value of l_{d} is equivalent to estimating the average SRO over a longer period of time, setting a trade-off between robust estimation and the ability to track time-varying SROs. Since fixed SROs are considered here, l_{d} may be safely set to a relatively large value, bearing in mind that SRO estimation can only begin after l_{d} frames. Temporal averaging can be applied before computing the phase to smoothen the estimation:

$$\bar{P}_{\Gamma,kq}[\kappa,i] = \alpha \bar{P}_{\Gamma,kq}[\kappa,i-1] + (1-\alpha) P_{\Gamma,kq}[\kappa,i], \quad (15)$$

where α is a scalar, $0 \ll \alpha \leq 1$, set close to 1.

Since (14) and (15) are defined for all frequency bins κ, the SRO can be estimated, for example, as the least squares (LS) solution over all relevant frequency bins [21]. Since this LS solution is, however, prone to inaccuracies due to the periodicity of the phase, it has been proposed in [12] to interpret $\bar{P}_{\Gamma,kq}[\kappa,i]$ as a generalised cross-PSD. The integer time lag $\lambda_{\max}[i]$ that maximises the absolute value of the generalised cross-correlation $\bar{p}^{i}_{\Gamma,kq}[\lambda] = \mathcal{F}^{-1}\{\bar{P}_{\Gamma,kq}[\kappa,i]\}$, with $\mathcal{F}^{-1}\{\cdot\}$ denoting the inverse DFT, can then be used to estimate the SRO as:

$$\hat{\varepsilon}_{kq}[i] = -\frac{\lambda_{\max}[i]}{l_{\mathrm{d}} N_{\mathrm{s}}} = -\frac{1}{l_{\mathrm{d}} N_{\mathrm{s}}} \cdot \arg\max_{\lambda} |\bar{p}^{i}_{\Gamma,kq}[\lambda]|. \quad (16)$$

Higher SRO estimation accuracy can be obtained by determining the non-integer value $\lambda[i]$ that maximises $|p^{i}_{\Gamma,kq}[\lambda]|$, via an interpolation method such as a golden section search in the interval $[\lambda_{\max}[i] - 0.5, \lambda_{\max}[i] + 0.5]$, as proposed in [12], and substituting $\lambda_{\max}[i]$ by $\lambda[i]$ in (16).

B. SRO compensation and full-sample drifts

The SRO estimates obtained via the method described in Section IV-A are now used to perform SRO compensation on the elements of $\mathbf{z}_{-k}[\kappa,i]$ before updating $\tilde{\mathbf{w}}_k[\kappa,i]$ as described in Section III. Using the LPD model, SRO compensation can be performed at any node k based on $\{\hat{\varepsilon}_{kq}[i]\}_{q\in\mathcal{K}_k}$ by applying the appropriate phase shift to each element of $\mathbf{z}_{-k}[\kappa,i]$ as:

$$\check{z}_q[\kappa,i] = z_q[\kappa,i] \cdot \exp\left(-\mathrm{j}\frac{2\pi}{N}\kappa\hat{\varphi}_{kq}[i]\right) \ \forall\, q \in \mathcal{K}_k, \quad (17)$$

where $\hat{\tau}_{kq}[i] = N_{\mathrm{s}} \sum_{\iota=0}^{i} \hat{\varepsilon}_{kq}[\iota]$ is the estimated average accumulated time-drift between $z_q[\kappa,i]$ and $y_{k1}[\kappa,i]$ (cfr. (11)).

An important aspect comes into play $t^{\mathrm{FSD}}_{kq} = 1/(f_{\mathrm{s},k}|\varepsilon_{kq}|)$ seconds after the simultaneous initialisation of the WASN, namely when the accumulated SRO-induced time drift $\tau_{kq}[i]$ between node k and node q becomes greater than one sample. Such event is referred to in the following as a full-sample drift (FSD). At that time, if $\varepsilon_{kq} > 0$, the growing time drift between node k and node q leads to a situation where node q has recorded one more sample than node k, as depicted in Figure 1. Conversely, if $\varepsilon_{kq} < 0$, node q has recorded one less sample than node k.

When correctly detected, an FSD can be compensated for by applying a corrective phase shift $\phi^{\mathrm{FSD}}_{kq}[\kappa,i]$ to $z_q[\kappa,i]$ as:

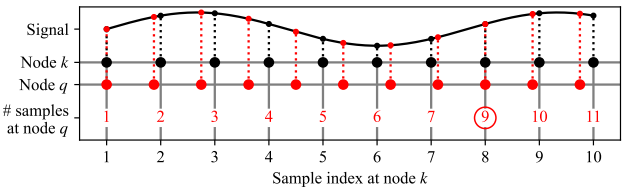

FIGURE 1. Schematic representation of a full-sample drift (indicated by the circle) generated by an SRO $\varepsilon_{kq} > 0$ between node k and $q \in \mathcal{K}_k$.

$$\phi^{\mathrm{FSD}}_{kq}[\kappa,i] = \begin{cases} \exp\left(-\mathrm{j}\frac{2\pi}{N}\kappa\right) & \text{if one more sample at } q, \\ \exp\left(\mathrm{j}\frac{2\pi}{N}\kappa\right) & \text{if one less sample at } q, \\ 1 & \text{otherwise.} \end{cases} \quad (18)$$

In order to continuously compensate for FSDs within the LPD framework, the phase shifts from (18) must be accumulated over time, defining $\phi^{\mathrm{ac}}_{kq}[\kappa,i]$ as:

$$\phi^{\mathrm{ac}}_{kq}[\kappa,i] = \phi^{\mathrm{ac}}_{kq}[\kappa,i-1] \cdot \phi^{\mathrm{FSD}}_{kq}[\kappa,i] = \prod_{\iota=0}^{i} \phi^{\mathrm{FSD}}_{kq}[\kappa,\iota]. \quad (19)$$

FSD compensation can be incorporated within the SRO estimation process by multiplying the PSD estimate $\Psi_{kq}[\kappa,i]$ by $\phi^{\mathrm{ac}}_{kq}[\kappa,i]$. Since the denominator of (12) is unaffected by this phase shift, this is equivalent to multiplying $P_{\Gamma,kq}[\kappa,i]$ by the product $\phi^{\mathrm{ac}}_{kq}[\kappa,i] \cdot \phi^{\mathrm{ac},*}_{kq}[\kappa,i-l_{\mathrm{d}}]$. The rest of the SRO estimation process is unchanged, following (15) and (16).

However, it should be realised that the accumulated effect of FSDs becomes particularly problematic when considering the WOLA implementation of DANSE [13], where a fused time-domain signal $\dot{z}_q[n]$ is transmitted in frames of N_{s} samples from node q to node k (cfr. Algorithm 1). For clarity of exposition, we assume an even DFT size N and a 50% WOLA window shift such that $N_{\mathrm{s}} = N/2$. A problematic phenomenon referred to as full-frame drift (FFD) occurs when N_{s} uncompensated FSDs accumulate. If $\varepsilon_{kq} > 0$ (resp. $\varepsilon_{kq} < 0$) and after $t^{\mathrm{FFD}}_{kq} = N_{\mathrm{s}} t^{\mathrm{FSD}}_{kq}$ seconds, node q has recorded N_{s} more (resp. less) samples than node k since the synchronous initialisation of both nodes. At that time, node q has thus transmitted *two* (resp. *no*) new $\dot{z}_q[n]$ frames since the last update of node k (see circles on Figure 2). Consequently, to perform its next update, node k *skips* (resp. *duplicates*) one $z_q[\kappa,i]$ frame.

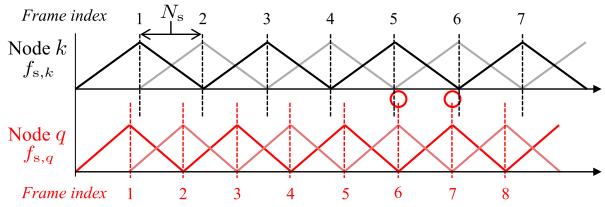

FIGURE 2. Schematic representation of a full-frame drift (highlighted by circles), with $\varepsilon_{kq} > 0$ and 50% WOLA window shift.

An FFD cannot be compensated for via a phase shift based on (18) if N_{s} is close to N. For instance, with 50%

WOLA window shift, the corrective phase shift of (18) needed to compensate for N_s FSDs at once simplifies to $\exp(\pm\mathrm{j}\frac{2\pi}{N}\kappa N_\mathrm{s}) = \pm 1\ \forall\kappa$. Even if FFDs compensation were possible, before an FFD occurs, node k receives a single N_s samples-long frame of $\dot{z}_q[n]$ between two consecutive local filter updates, as shown in Figure 2. Node k is, therefore, unable to detect FSDs by comparing the number of local $\dot{\mathbf{y}}_k[n]$ samples with the number of received $\dot{z}_q[n]$ samples since its previous update (as both are equal to N_s). The uncompensated growing drift between elements of $\tilde{\mathbf{y}}_k[\kappa, i]$ then leads to increasingly erroneous updates of the covariance matrices.

If disregarded, FFDs can significantly perturb the convergence of DANSE as well as the SRO estimation process from Section IV-A. The mitigation of FFDs and the detection of FSDs within the WOLA implementation of DANSE is discussed in the following section.

C. Full-sample drift detection

In order to enable detection and compensation of FSDs within the WOLA implementation of DANSE, we introduce a modification of the DANSE fusion and broadcasting mechanism to allow per-sample transmission of fused signals between nodes, while retaining WOLA frame-by-frame processing for the computationally costly steps of GEVD-based filter update and desired signal estimate computation. In principle, using this per-sample transmission, node k can easily detect FSDs for the i-th filter update by comparing the number of local $\dot{\mathbf{y}}_k[n]$ samples with the number of received $\dot{z}_q[n]$ samples from node q since its previous update, then compensate for them via the corrective phase shifts of (18).

We propose to approximate the WOLA filtering process (analysis, STFT-domain filtering, and synthesis) by its so-called distortion function $T(\zeta)$ [22], where ζ is the $\mathcal{Z}$-transform variable. This function relates the output of the WOLA filterbank to its input when no up- and downsampling is performed, i.e., using maximal window overlap. At frame i, the distortion function $T^i_{q,m}(\zeta)$ corresponding to the m-th microphone of node q can be obtained as:

$$T^i_{q,m}(\zeta) = \frac{1}{N_\mathrm{s}}\left[\zeta^{1-N}\ \dots\ 1\right]\mathbf{D}^i_{q,m}\left[1\ \dots\ \zeta^{1-N}\right]^T, \quad (20)$$

with $\mathbf{D}^i_{q,m} = \mathbf{H}_\mathrm{s}\cdot\mathbf{F}^{-1}\cdot\mathrm{diag}\{\mathbf{w}_{qq,m}[i]\}\cdot\mathbf{F}\cdot\mathbf{H}_\mathrm{a}$, where $\mathbf{F}^{-1}$ and $\mathbf{F}$ are the inverse DFT and DFT matrix, respectively, $\mathbf{w}_{qq,m}[i] = \left[w_{qq,m}[1,i],\dots,w_{qq,m}[N,i]\right]^T$ denotes the local filter coefficients at frame i for the m-th microphone of node q with all frequency bins stacked into one vector, $\mathbf{H}_\mathrm{s} = \mathrm{diag}\{\mathrm{flip}\{\mathbf{h}_\mathrm{s}\}\}$, and $\mathbf{H}_\mathrm{a} = \mathrm{diag}\{\mathbf{h}_\mathrm{a}\}$, respectively, where $\mathbf{h}_\mathrm{s}$ and $\mathbf{h}_\mathrm{a}$ denote the WOLA synthesis and analysis time-domain windows, respectively, and $\mathrm{flip}\{\cdot\}$ reverses the order of the elements of a vector.

The time-domain equivalent of the distortion function $T^i_{q,m}(\zeta)$ in (20) is a $(2N-1)$-tap impulse response denoted by $\mathbf{t}^i_{q,m}$. From (20), it can be seen that each element of $\mathbf{t}^i_{q,m}$ is obtained by summing over the corresponding diagonal of the matrix $\mathbf{D}^i_{q,m}$. The complete WOLA analysis and synthesis process can then be approximated by a convolution with $\mathbf{t}^i_{q,m}$. This means that the n-th sample of the time-domain fused signal $\dot{z}_q[n]$ can be obtained as:

$$\dot{z}_q[n] = \sum_{m=1}^{M_q}\left(\dot{\mathbf{y}}^{(n)}_{q,m} * \mathbf{t}^i_{q,m}\right)[n+2N-1] \quad (21)$$

where the time-domain vector $\dot{\mathbf{y}}^{(n)}_{q,m}$ contains the most recent N samples recorded by the m-th microphone of node q and $(\mathbf{a}*\mathbf{b})[c]$ denotes the c-th sample of the convolution between vectors $\mathbf{a}$ and $\mathbf{b}$. Note that the distortion function does not need be computed at every frame i, especially once the filters have converged after several DANSE iterations. The iteration indices at which the distortion function is updated with the most recent filter $\mathbf{w}_{qq}[\kappa, i]$ are grouped in the set $\mathcal{I}_\mathrm{T}$.

Although the proposed $T(\zeta)$-approximation introduces the same $N-1$ samples input-output delay as the standard WOLA implementation of DANSE [13], it has the advantage to circumvent the N_s samples delay introduced by frame-by-frame processing [18] since no downsampling is performed.

Using the $T(\zeta)$-approximation, any node is now able to broadcast its fused signal on a per-sample basis. This modification of the DANSE algorithm, although coming at the expense of some additional computational complexity with respect to a purely WOLA-based implementation, enables the deployment of DANSE in asynchronous WASNs where FSDs can be detected as soon as they occur. An overview of the DANSE algorithm at node $k \in \mathcal{K}$ with per-sample fused signal broadcasting using the $T(\zeta)$-approximation is provided in Figure 3. SRO estimation and compensation is not depicted in this figure.

D. Complete system

As SRO estimation is necessary for SRO compensation, both should be performed in parallel. An open-loop strategy is proposed here, depicted in Figure 4 for node $k \in \mathcal{K}$, which consists of three parts: (i) SRO estimation, (ii) FSD detection and compensation, and (iii) SRO compensation. First, the original (SRO-uncompensated) fused signal $z_q[\kappa, i]$ is used to estimate $\hat{\varepsilon}_{kq}[i]$. Every time an FSD is detected (i.e., a mismatch in the number of locally recorded samples and received samples since the last filter update, cfr. Section IV-C), a flag is raised and the accumulated phase shift of (19) is applied to the appropriate parts of the system. SRO compensation is then applied, leading to the signal $\check{z}_q[\kappa, i]$, which is used to update the DANSE filter.

Algorithm 2 provides a complete description of WOLA-based DANSE with SRO estimation and compensation, including the $T(\zeta)$-approximation for FSD detection. There, the STFT-domain SRO-compensated fused signals vector is denoted by $\check{\mathbf{z}}_{-k}[\kappa, i]$ and the SRO-compensated version of $\tilde{\mathbf{y}}_k[\kappa, i]$ becomes $\check{\mathbf{y}}_k[\kappa, i] = [\mathbf{y}^T_k[\kappa, i] \mid \check{\mathbf{z}}^T_{-k}[\kappa, i]]^T$. The estimates of $E\{\check{\mathbf{y}}_k[\kappa, i]\check{\mathbf{y}}^H_k[\kappa, i]\}$ and $E\{\check{\mathbf{n}}_k[\kappa, i]\check{\mathbf{n}}^H_k[\kappa, i]\}$ are denoted by $\check{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}[\kappa, i]$ and $\check{\mathbf{R}}_{\mathbf{n}_k\mathbf{n}_k}[\kappa, i]$, respectively.

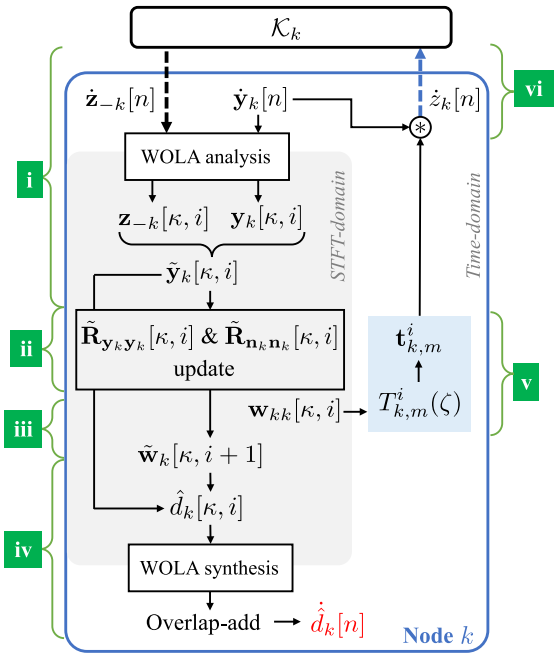

FIGURE 3. **Proposed WOLA-based DANSE processing at node $k \in \mathcal{K}$ with per-sample fused signal broadcasting. [i] WOLA analysis applied to local microphone signals $\dot{\mathbf{y}}_k[n]$ and fused signals from other nodes $\dot{\mathbf{z}}_{-k}[n]$. [ii] Covariance matrix update and [iii] computation of filter $\tilde{\mathbf{w}}_k[\kappa,i+1]$. [iv] Computation of new desired signal estimate frame $\hat{d}_k[\kappa,i+1]$, followed by WOLA synthesis and overlap-add. [v] Computation of istortion functions $\{T^i_{k,m}(\zeta)\}_{m=1}^{M_k}$ from filter $\mathbf{w}_{kk}[\kappa,i]$. [vi] Computation of new $\dot{z}_k[n]$ samples and per-sample broadcasting to other nodes.**

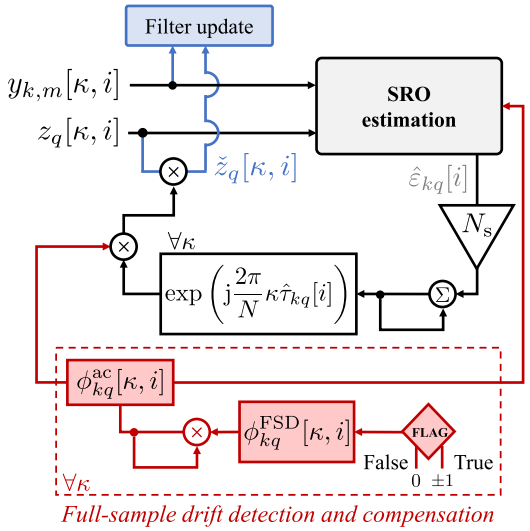

FIGURE 4. **SRO estimation and compensation block-scheme at node $k \in \mathcal{K}$, including full-sample drift detection ("flag") and compensation.**

The filter estimate after SRO compensation is finally obtained similarly to (9), i.e., performing a GEVD on the matrix pencil $\{\check{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}[\kappa,i], \check{\mathbf{R}}_{\mathbf{n}_k\mathbf{n}_k}[\kappa,i]\}$, and is denoted by $\check{\mathbf{w}}_k[\kappa,i+1]$.

Algorithm 2 WOLA-based DANSE with SRO compensation in a fully connected heterogeneous WASN.

Initialise $\tilde{\mathbf{w}}_k[\kappa,0]\ \forall\,(k,\kappa) \in \mathcal{K}\times\{1,\dots,N\}$;
Each node $k \in \mathcal{K}$ performs the following, starting simultaneously;
for every new locally recorded sample $\dot{\mathbf{y}}[n]$ **do**
  Compute $\dot{z}_k[n]$ via (21) and transmit to nodes in $\mathcal{K}_k$.
end for
for $i = 1, 2, 3, \dots$ **do**
  if $i \in \mathcal{I}_{\mathrm{T}}$ **then**
    Update $\{T^i_{k,m}(\zeta)\}_{m=1}^{M_k}$ via (20) using $\mathbf{w}_{kk}[\kappa,i]$;
  else
    $\{T^i_{k,m}(\zeta)\}_{m=1}^{M_k} = \{T^{i-1}_{k,m}(\zeta)\}_{m=1}^{M_k}$;
  end if
  Shift WOLA window (N_{s} new samples since $i-1$);
  WOLA analysis on local signals to obtain $\mathbf{y}_k[\kappa,i]$;
  WOLA analysis on fused signals to obtain $\mathbf{z}_{-k}[\kappa,i]$;
  for $q \in \mathcal{K}_k$ **do**
    Detect FSDs based on number of new $\dot{z}_q[n]$ samples and compute $\phi^{\mathrm{ac}}_{kq}[\kappa,i]\ \forall\,\kappa$ via (18) and (19);
    Compute $\hat{\varepsilon}_{kq}[i]$ via (16);
    Compute $\check{z}_q[\kappa,i]$ (Figure 4) and build $\check{\mathbf{z}}_{-k}[\kappa,i]$;
  end for
  Build $\check{\mathbf{y}}_k[\kappa,i] = [\mathbf{y}_k^T[\kappa,i] \mid \check{\mathbf{z}}_{-k}^T[\kappa,i]]^T$;
  Compute $\check{\mathbf{R}}_{\mathbf{y}_k\mathbf{y}_k}[\kappa,i]$ and $\check{\mathbf{R}}_{\mathbf{n}_k\mathbf{n}_k}[\kappa,i]$;
  Compute $\check{\mathbf{w}}_k[\kappa,i+1]$ via (9), then $\hat{d}_k[\kappa,i]$.
  WOLA synthesis on $\hat{d}_k[\kappa,i]$ and overlap-add with the previous frame to build $\dot{\hat{d}}_k[n]$.
end for

V. Numerical experiments

The performance of Algorithm 2 is demonstrated via numerical experiments and compared to Algorithm 1. The acoustic environment layout is depicted in Figure 5. A WASN consisting of $K = 4$ nodes is considered, with $\{M_k\}_{k=1}^4 = \{1,3,2,5\}$ microphones arranged in grid arrays with a 20 cm inter-microphone spacing. A room with dimensions 5×5×5 m^3 is considered, with a uniform absorption coefficient of 0.9 resulting in a reverberation time of $T_{60} = 0.15$ s. One localised target speech source and two localised uncorrelated stationary white noise sources are present. The speech signal consists of 3 s long snippets from LibriSpeech [23], starting with 0.25 s of silence and each snippet separated by 2 s of silence. The power of each source is set to obtain an SNR of -3 dB at the reference microphone of node 1. All signals are 15 s long, where the microphone signals are simulated by convolving the source signals with 4096 samples-long room impulse responses obtained using the randomised image method [24]. The nominal sampling rate is set to 16 kHz.

The filters at each node are initialised as merely selecting the local reference microphone signal, i.e., $\tilde{\mathbf{w}}_k[\kappa,0] = [1\ \mathbf{0}]^T\ \forall\ k \in \mathcal{K}$. The covariances matrices are updated using a forgetting factor $\beta = 0.978$ (cfr. (4)). All WOLA processing is performed using an $N = 1024$ samples square-

root Hann window for both analysis and synthesis and 50% overlap. FSDs are detected using the WOLA approximation described in Section IV-C, where the distortion function $T^i_{k,m}(\zeta)$ in (20) is updated based on the filter $\mathbf{w}_{kk}[\kappa, i]$ every 30 DANSE iterations. The covariance matrices are estimated via (4) assuming an ideal VAD, which avoids the influence of VAD errors on the results. In practice, the VAD obviously needs to be estimated from the microphone signals [14], [15].

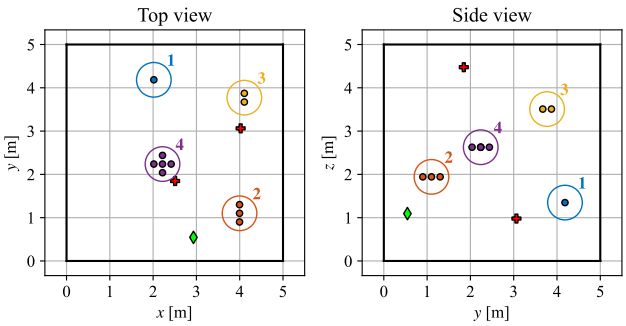

FIGURE 5. Layout of acoustic scenario used in the simulations. Microphones (○) are grouped in nodes (◯) numbered 1 to 4. Two noise sources (+) and one desired source (◇) are present.

The clock of node 1 is systematically set as the reference clock with a sampling rate exactly equal to 16 kHz. The SRO for all other nodes $k \in \{2, 3, 4\}$ is set with respect to this reference. Three degrees of network asynchronicity are considered based on the measured SROs reported in [5]. First, small SROs are considered by setting $\{\varepsilon_{1k}\}_{k=2}^4 = \{20, -20, 40\}$ PPM. Second, more asynchronicity is applied by setting $\{\varepsilon_{1k}\}_{k=2}^4 = \{50, -50, 100\}$ PPM. Finally, a strongly asynchronous network is simulated by setting $\{\varepsilon_{1k}\}_{k=2}^4 = \{200, -200, 400\}$ PPM. Fixed SROs are simulated at any node by resampling the signals appropriately. The SROs estimation method described in Section IV-A uses $l_\mathrm{d} = 10$ in (13) and $\alpha = 0.95$ in (15).

The performance at each node is quantified using the extended short-term objective intelligibility (eSTOI) [25], which uses the clean speech component of the local reference microphone as reference. The metric is computed on the signal segment starting from WOLA frame $i = 15$ (around $t = 0.5$ s) to reduce the impact of initial filter updates. For the three considered degrees of asynchronicity, Figure 6 shows the eSTOI values at each node for (i) the local reference microphone signal (without any noise reduction), (ii) the desired signal estimate obtained using Algorithm 1 without SRO compensation, and (iii) the desired signal estimate obtained using Algorithm 2 with the proposed SRO compensation. The eSTOI obtained using the centralised GEVD-MWF (cfr. (6)) without any asynchronicity in the WASN is provided for comparison.

The results show that the presence of SROs in the WASN significantly deteriorates the performance of WOLA-based GEVD-DANSE. The single-microphone node ($M_1 = 1$) is particularly sensitive to the presence of SROs as it heavily

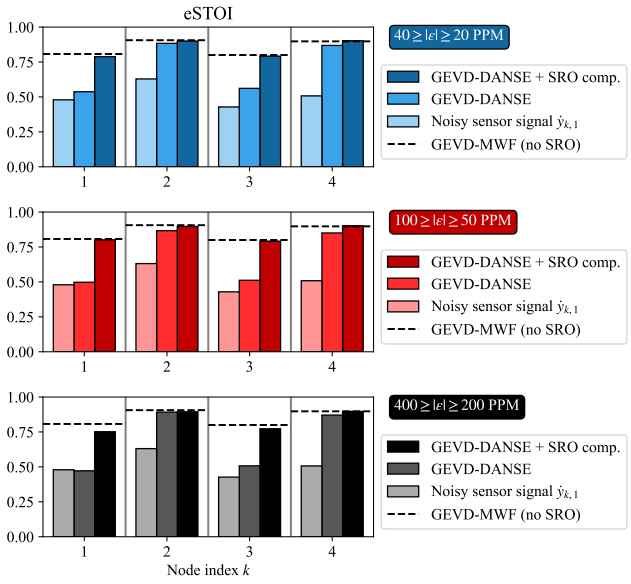

FIGURE 6. eSTOI obtained at each node of the WASN depicted in Figure 5 with small SROs (top), moderate SROs (middle), or large SROs (bottom). Local noisy reference microphone signals without any processing (lightest), GEVD-DANSE estimates in the presence of SROs (light), GEVD-DANSE estimates with SRO estimation and compensation (darker), and GEVD-MWF SRO-free centralised estimates (dashed).

relies on the information provided by other nodes to compute its desired signal estimate. This occurs regardless of the considered SRO magnitude, which shows the negative impact of even relatively small SROs. Conversely, nodes including many microphones (e.g., $M_4 = 5$), show almost no sensitivity to SROs, suggesting that these nodes may be able to mainly rely on their local observations to perform noise reduction. For all considered SRO magnitudes, each node using the proposed SRO estimation and compensation method is able to restore the centralised performance that GEVD-DANSE would showcase in an SRO-free WASN.

VI. Conclusion

In this contribution, the WOLA-based implementation of the GEVD-DANSE algorithm has been rendered robust to the presence of SROs by combining a coherence-based SRO estimation technique with an approximation of the WOLA process to allow FSDs detection and compensation via per-sample broadcasting of fused signals. The performance of the proposed method has been assessed through numerical experiments in the context of speech enhancement. in terms of intelligibility of the desired signal estimate at each node. The results show that even relatively small SROs (if not estimated and compensated for) can have a detrimental impact on the ability of DANSE to recover the desired signal at nodes that significantly rely on collaboration with other nodes. However, it is shown that, in an asynchronous WASN, the proposed SRO estimation and compensation method practically restores the performance that the GEVD-DANSE algorithm would showcase in a fully synchronised network.

REFERENCES

[1] A. Bertrand, "Applications and trends in wireless acoustic sensor networks: A signal processing perspective," in *Proc. 18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*, Ghent, Belgium, 2011, pp. 1–6.

[2] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks—Part I: Sequential node updating," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5277–5291, 2010.

[3] A. Bertrand and M. Moonen, "Distributed adaptive node-specific signal estimation in fully connected sensor networks—Part II: Simultaneous and asynchronous node updating," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5292–5306, 2010.

[4] S. Ruiz, T. van Waterschoot, and M. Moonen, "Distributed combined acoustic echo cancellation and noise reduction in wireless acoustic sensor and actuator networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 30, pp. 534–547, 2022.

[5] M. Guggenberger, M. Lux, and L. Böszörmenyi, "An analysis of time drift in hand-held recording devices," in *MultiMedia Modeling*, vol. 8935, pp. 203–213. Springer International Publishing, 2015.

[6] R. Lienhart, I. Kozintsev, S. Wehr, and M. Yeung, "On the importance of exact synchronization for distributed audio signal processing," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Hong Kong, China, 2003, vol. 4, pp. IV–840–3.

[7] A. Hassani, A. Bertrand, and M. Moonen, "GEVD-based low-rank approximation for distributed adaptive node-specific signal estimation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 64, no. 10, pp. 2557–2572, 2016.

[8] J. Szurley, A. Bertrand, and M. Moonen, "Topology-independent distributed adaptive node-specific signal estimation in wireless sensor networks," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 1, pp. 130–144, 2017.

[9] R. Van Rompaey and M. Moonen, "Distributed adaptive signal estimation in wireless sensor networks with partial prior knowledge of the desired sources steering matrix," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 7, pp. 478–492, 2021.

[10] J. Schmalenstroeer, J. Heymann, L. Drude, C. Boeddecker, and R. Haeb-Umbach, "Multi-stage coherence drift based sampling rate synchronization for acoustic beamforming," in *Proc. 2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*, Luton, 2017, pp. 1–6.

[11] A. Chinaev, P. Thuene, and G. Enzner, "Double-cross-correlation processing for blind sampling-rate and time-offset estimation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 1881–1896, 2021.

[12] T. Gburrek, J. Schmalenstroeer, and R. Haeb-Umbach, "On synchronization of wireless acoustic sensor networks in the presence of time-varying sampling rate offsets and speaker changes," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2022, pp. 916–920.

[13] A. Bertrand and M. Moonen, "Robust distributed noise reduction in hearing aids with external acoustic sensor nodes," *EURASIP J. Adv. Signal. Process.*, vol. 2009, no. 1, pp. 530435, 2009.

[14] A. Bertrand and M. Moonen, "Energy-based multi-speaker voice activity detection with an ad hoc microphone array," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Dallas, TX, USA, 2010, pp. 85–88, IEEE.

[15] Y. Zhao, J. K. Nielsen, J. Chen, and M. G. Christensen, "Model-based distributed node clustering and multi-speaker speech presence probability estimation in wireless acoustic sensor networks," *J. Acoust. Soc. Am.*, vol. 147, no. 6, pp. 4189–4201, 2020.

[16] R. Serizel, M. Moonen, B. Van Dijk, and J. Wouters, "Low-rank approximation based multichannel Wiener filter algorithms for noise reduction with application in cochlear implants," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 4, pp. 785–799, 2014.

[17] J. Szurley, A. Bertrand, and M. Moonen, "Improved tracking performance for distributed node-specific signal enhancement in wireless acoustic sensor networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2013, pp. 336–340.

[18] R. Crochiere, "A weighted overlap-add method of short-time Fourier analysis/Synthesis," *IEEE Trans. Audio, Speech, Language Process.*, vol. 28, no. 1, pp. 99–102, 1980.

[19] S. Miyabe, N. Ono, and S. Makino, "Blind compensation of inter-channel sampling frequency mismatch with maximum likelihood estimation in STFT domain," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2013, pp. 674–678.

[20] L. Wang and S. Doclo, "Correlation maximization-based sampling rate offset estimation for distributed microphone arrays," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 3, pp. 571–582, 2016.

[21] M. Bahari, A. Bertrand, and M. Moonen, "Blind sampling rate offset estimation for wireless acoustic sensor networks through weighted least-squares coherence drift estimation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 3, pp. 674–686, 2017.

[22] P. Vaidyanathan, *Multirate systems and filter banks*, Prentice Hall P T R, Englewood Cliffs, New Jersey 07632, 1993.

[23] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2015, pp. 5206–5210.

[24] E. De Sena, N. Antonello, M. Moonen, and T. van Waterschoot, "On the modeling of rectangular geometries in room acoustic simulations," *IEEE Trans. Audio, Speech, Language Process.*, vol. 23, no. 4, pp. 774–786, 2015.

[25] J. Jensen and C. Taal, "An algorithm for predicting the intelligibility of speech masked by modulated noise maskers," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 11, pp. 2009–2022, 2016.